gemalto
security to be free

HSMoD Service

MICROSOFT AUTHENTICODE INTEGRATION GUIDE

Document Information

Product Version	1.7
Document Part Number	007-013897-001
Release Date	21 November 2018

Revision History

Revision	Date	Reason
Rev. A	17 August 2017	For initial release 1.1.0
Rev. B	19 September 2017	For release 1.1.1
Rev. C	14 November 2017	For release 1.2
Rev. D	05 February 2018	For release 1.3
Rev. E	02 March 2018	For HSM on Demand release 1.3
Rev. F	05 April 2018	For release 1.4
Rev. G	07 May 2018	For HSM on Demand release 1.4
Rev. H	10 June 2018	For release 1.5
	12 September 2018	For release 1.6
	21 November 2018	For release 1.7

The information contained in this document is provided “AS IS” without any warranty of any kind. Unless otherwise expressly agreed in writing, Gemalto makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Gemalto reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Gemalto hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Gemalto be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Gemalto does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Gemalto be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Gemalto products. Gemalto disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service, or loss of privacy.

HSM on Demand1.7

MICROSOFT AUTHENTICODE INTEGRATION GUIDE

Contents

Overview

This document covers the necessary information to install, configure, and integrate Microsoft Authenticode with an HSM on Demand Service.

Authenticode relies on proven cryptographic techniques and the use of one or more private keys to sign and time-stamp the published software. It is important to maintain the confidentiality of these keys. HSMoD Service integrates with Microsoft Authenticode to provide a trusted system for protecting the organizational credentials of the software publisher. An HSM on Demand Service secures the code-signing key within an industry standard FIPS 140-2 level 3 validated HSM.

This document contains the following sections:

Third Party Application Details

This integration guide uses the following third party applications:

> Microsoft Authenticode (Microsoft Windows SDK 10.1)

Supported Platforms

The following platforms are tested with HSM on Demand Service:

Platforms Tested	Microsoft SDK	Microsoft Office Smart Tags SDK (optional)
Windows Server 2016	10.1	Office 2003 SDK
Windows Server 2012 R2	10.1	Office 2003 SDK

NOTE Microsoft Authenticode Integration is tested with Luna Clients in both FIPS and HA mode.

Preparing for the Integration

Before you proceed with the integration, ensure you have completed the following:

> "Provision HSM on Demand Services" below
> "Install Windows SDK" on page 9

Provision HSM on Demand Services

The HSM on Demand Service provides your client machine with access to an HSM application partition for storing cryptographic objects used by your applications. Application partitions can be assigned to a single client, or multiple clients can be assigned to, and share, a single application partition.

You must provision your HSM on Demand service by adding the service, downloading the service client package and initializing the HSM. Provisioning your HSM on Demand service entails:

> "Adding a Service" below
> "Adding a Service Client" on the next page
> "Initializing the HSM" on page 8

Adding a Service

1. Under the **Services** tab, select the **Add New Service** page. Click **Deploy** on the service tile for the service you wish to add.

   NOTE Click **Deploy** on the HSM on Demand Service tile for your integration.

2. Review the "Terms of Services DPoD." Enable the **I have read and accept the Terms of Service above** check box and then click **Next**.
3. On the **Add <service_type> Service** page, enter a name for the Service in the **Service Name** field. You can optionally allow non-FIPS approved algorithms by selecting the **Allow non-FIPS approved algorithms** check box. Click **Next**.

   CAUTION! You cannot alter the FIPS setting after creating the service. You must decide if the service should allow or disallow non-FIPS approved algorithms before clicking **Finish** in the next step.

4. Review the configuration summary page. If acceptable, click **Finish**. If you would like to make changes to the configuration, click **Go Back**.

   When completed, the new service is listed under **My Services** and a **Create Service Client?** window displays.

5. Click **Create Service Client**.

Adding a Service Client

1. In the **Create Service Client** window enter a name for the service client in the **Service Client Name** field.

   NOTE If the **Create Service Client** window is not available, navigate to the **Services** tab and click the name of the Service you would like to generate a client for in the **My Services** table. On the Service Details page, click **New Service Client**.

2. Select **Create Service Client**.

   A new HSM service client package is created and provided for downloading on your client system.

   NOTE The HSM service client package is a zip file that contains system information needed to connect your client system to an existing HSM on Demand service. The HSM service client package should download immediately on creation. If it does not, or you lose access to your HSM service client package it can be accessed or reacquired through the **My Services** table.

3. Transfer the service client package to your client system. You can use SCP, PSCP, WinSCP, FTPS, or any other secure file transfer tool.
4. Unzip the service client package.

   For Linux, enter:

   ```
   unzip <service_client_package>.zip
   ```

   For Windows, using the Windows GUI or an unzip tool unzip the file:

   ```
   <service_client_package>.zip
   ```

   NOTE For more information about the service client package contents see .

5. Extract the cvclient-min file.

   NOTE Extract the cvclient-min file in the directory where you extracted the <service_client_package>.zip. **Do not** extract to a new cvclient-min directory.

   For Linux, untar the cvclient-min.tar

   ```
   tar xvf cvclient-min.tar
   ```

   For Windows, unzip the cvclient-min.zip.
6. Set the environment variable.

   For Linux, execute:

   ```
   source ./setenv
   ```

   For Windows, right click setenv.cmd and select **Run as Administrator**.

NOTE If you encounter the error dll load failed with GetLastError() 126 move the contents of the cvclient_min folder up one directory and execute setenv.

7. Start LunaCM.

   For Linux, execute the following from the directory where you extracted the cvclient-min.tar file.

   ```
   ./bin/64/lunacm
   ```

   For Windows, execute the following from the directory where you unzipped the cvclient-min.zip file.

   ```
   lunacm
   ```

Initializing the HSM

1. Set the active slot to the service partition.

   ```
   lunacm:>slot set -slot <slot_number>
   ```

   NOTE Execute slot list in LunaCM to identify the slot number associated with your service.

2. Initialize the application partition. During this process you will create the partition's Security Officer (SO), set the SO password, and specify the cloning domain.

   ```
   lunacm:> partition init -label <service_label>
   ```

3. Optional: If you wish to transfer key material to or from a PED-authenticated Luna partition, you initialize the SafeNet Data Protection On Demand partition using the red PED domain key.

   a. For DPoD deployments, contact customer support to obtain the necessary PED drivers so that your HSM client can communicate with the PED.

   b. Attach the PED locally to the client computer, insert the red cloning domain PED key, and initialize the partition, including the option to set the cloning domain from the red PED key. Execute:

   ```
   lunacm:> partition init -label <cryptovisor_partition_label> -importpeddomain
   ```

4. Log in as the partition's Security Officer:

   ```
   lunacm:>role login -name Partition SO
   ```

5. Initialize the Crypto Officer role:

   ```
   lunacm:>role init -name Crypto Officer
   ```

6. Log out of the partition Security Officer role and log in as the Crypto Officer.

   ```
   lunacm:>role logout
   lunacm:>role login -name Crypto Officer
   ```

7. You must change the Crypto Officer password immediately on the initial log in. Failure to due so will result in a password error on subsequent logins.

   ```
   lunacm:>role changepw -name Crypto Officer
   ```

8. Initialize the Crypto User role:

   ```
   lunacm:>role init -name Crypto User
   ```

9. Log out of the partition Crypto Officer role and log in as the Crypto User.

```
lunacm:>role logout
lunacm:>role login -name Crypto User
```

10. You must change the Crypto User password immediately on the initial log in. Failure to do so will result in a password error on subsequent logins.

    ```
    lunacm:>role changepw -name Crypto User
    ```

    This completes initializing the HSM on Demand Service. The Crypto Officer and Crypto User roles can now be used to integrate applications with the HSMoD service to perform cryptographic operations

Constraints on HSMoD Services

Please take the following limitations into consideration when integrating your application software with an HSM on Demand Service.

HSM on Demand Service in FIPS mode

HSMoD services operate in a FIPS and non-FIPS mode. If your organization requires non-FIPS algorithms for your operations, ensure you enable the **Allow non-FIPS approved algorithms** check box when configuring your HSM on Demand service. The FIPS mode is enabled by default.

Refer to the *Mechanism List* in the SDK Reference Guide for more information about available FIPS and non-FIPS algorithms.

Verify HSM on Demand <slot> value

LunaCM commands work on the current slot. If there is only one slot, then it is always the current slot. If you are completing an integration using HSMoD services, you need to verify which slot on the HSMoD service you send commands to. If there is more than one slot, then use the **slot set** command to direct a command to a specified slot. You can use **slot list** to determine which slot numbers are in use by which HSMoD service.

Install Windows SDK

MS Authenticode requires additional libraries to integrate with HSMoD Service. You must install the following in addition to MS Authenticode:

- Microsoft Visual Studio
- Microsoft Windows SDK
- Microsoft Office Smart Tags SDK

Refer to the *Microsoft Windows SDK Installation* documentation and the *Microsoft Office Smart Tags SDK Installation* documentation for more information about installing the additional libraries.

Integrating Microsoft Authenticode with an HSM on Demand Service on Windows Server 2016 or Windows Server 2012 R2

Microsoft Authenticode permits end users to identify who published a software component and verify that no one has tampered with the software component before downloading the software component from the Internet.

This integration contains the following topics:

Installing the SafeNet Cryptographic Service Provider (CSP)

To use Microsoft Authenticode with an HSM on Demand Service you must configure the SafeNet Cryptographic Service Provider to generate the certificates for Microsoft Authenticode.

To install the SafeNet Cryptographic Service Provider

1. Install the SafeNet Cryptographic Service Provider (CSP):
   a. Navigate to the HSM client installation directory and execute:

      \CSP>register.exe

   b. Register the SafeNet Cryptographic Service Provider for Microsoft Windows.

      \CSP>register.exe /l

      NOTE You may need to transfer your **LunaCSP.dll** file to the CSP folder.

2. Open a Developer Command Prompt for Visual Studio and generate a certificate using SafeNet Cryptographic services.

   C:> makecert –sk <key_container_name> **-sp** <API_provider_name> **–r –n CN=**<certificate_name> **-ss** <name_of_certificate_store> <certificate_name>**.cer**

Signing and time stamping the code using the signtool GUI

You can use the signtool GUI wizard to sign and time stamp the code.

To sign and time stamp the code using the signtool GUI

1. Create a Software Publishing Certificate (SPC) using the recently generated certificate.

   Cert2Spc <certificate_name>**.cer** <certificate_name>**.spc**

2. Sign and time stamp the certificate using the signtool.
   a. Navigate to the Microsoft Platform SDK directory **C:\Program Files\Microsoft SDKs\Windows**<version>**\Bin**.
   b. Execute **signtool signwizard**.
   c. Click **Next**.
   d. Select the file to sign and click **Next**.
   e. Select **Custom** in the **Signing Options** window and click **Next**.
   f. Click **Select from File**. Select the generated Software Publishing Certificate **.spc**. Click **Next**.
   g. Select **Private Key in a CSP**. Select the CSP and Key Container. Click **Next**.
   h. Select the **Desired Hash Algorithm.** Click **Next**.
   i. Click **Next**.
   j. Add a description to the **Data Description** window, if desired. **Click** Next.
   k. Select **Add a timestamp to the data**. Provide the time stamping URL. Click **Next**.
   l. Click **Finish**.
3. Click **OK**.

> **NOTE** You can use the signtool wizard without accessing the gui interface. The following is an example of the command: **C:\> signtool sign /v /s** <name_of_certificate_store> **/csp "Cryptographic Service Provider Name" /kc** <key_container_name> **/t** <timestamp_URL> <file_to_be_signed>.

Signing and time stamping the code using the command line signtool

You can sign and time stampe the code using the command line **signtool**.

To sign and time stamp the code using the command line signtool

1. Navigate to the directory where signtool is stored.
2. Execute the following command:

   signtool sign /v /f <publisher_certificate> **/p** <HSM_partition_password> **/csp** <cryptographic service provider> **/k** <key_container_name> **/t** <timestamp_url> <file_to_be_signed>

Integrating MS Strong Name with an HSM on Demand Service on Windows Server 2012 R2

Strong Name is the part of Microsoft SDK that offers a powerful mechanism for giving .NET Framework assemblies unique identities. To get a valid strong name, an assembly is strong-name signed during the build process. This is done using the private key that corresponds to the public key in the strong name. The strong name signature can then be verified using the public key.

This integration contains the following topics:

Installing the SafeNet Cryptographic Service Provider (CSP)

To use MS Strong Name with an HSM on Demand Service you must configure the SafeNet Cryptographic Service Provider to generate the keys and certificates for MS Strong Name.

To install the SafeNet Cryptographic Story Provider (CSP)

1. Install the SafeNet Cryptographic Service Provider (CSP).
   a. Navigate to the lunaclient installation directory and execute:

      \CSP>register.exe

   b. Register the SafeNet Cryptographic Service Provider for Microsoft Windows.

      \CSP>register.exe /1
2. Generate a certificate using SafeNet Cryptographic services.

   C:> makecert –sk <key_container_name> **-sp** <API_provider_name> **–r –n CN=**<certificate_name> **-ss** <name_of_certificate_store> <certificate_name>**.cer**
3. Set the SafeNet CSP to be the default CSP for the system.

   sn.exe -c <API_provider_name>
4. Extract the public key from the key-pair generated in step 2 using the following command:

   sn.exe -pc <key_container_name> <public_key>

Signing a .NET Assembly

You can use MS Strong Name to sign any .NET assembly.

To sign a .NET assembly

1. Use the MS Visual Studio command prompt to compile the program and delay signing the generated .exe file.

   csc /delaysign+ /<public_key>**:"**<path_to_public_key> **C:\Users\Administrator\Desktop\myapp.cs**

2. Sign the key.

   sn.exe -Rc C:\Users\Administrator\Desktop\myapp.exe <key_container_name>

3. Verify the signature on the .NET assembly.

   sn.exe -v C:\Users\Administrator\Desktop\myapp.exe

Integrating Microsoft HCK with an HSM on Demand Service on Windows Server 2012

Microsoft's Windows Certification Program is designed to help your company deliver compatible and reliable systems, software, and hardware products. End users trust the logo as an assurance of compatibility and reliability. This program is intended to help you develop systems and devices that have been tested to ensure that they meet Microsoft standards for Windows 8.1 as well as the quality level that ensures a great Windows experience for end users.

- An HSM on Demand Service is used to secure the signing keys so that your signing keys are never accessed by any unauthorized entity. Microsoft HCK uses the RSA keys for signing the packages.
- Microsoft HCK is a 32 bit application so you have to use the Luna Clients with 32 bit CSP.

This integration contains the following topics:

Installing the SafeNet Cryptographic Service Provider (CSP)

To use Microsoft HCK with an HSM on Demand Service you must configure the SafeNet Cryptographic Service Provider to generate the certificates for Microsoft HCK.

To install the SafeNet Cryptographic Service Provider (CSP)

1. Install the SafeNet Cryptographic Service Provider (CSP).
   a. Navigate to the HSM client installation directory and execute:

      \CSP>register.exe

   b. Register the SafeNet Cryptographic Service Provider for Microsoft Windows.

      \CSP>register.exe /1
2. Verify the registered cryptographic providers.
3. Navigate to **C:\Windows\SysWOW64**.
4. Execute the command **certutil -csplist**.

Generating the Microsoft HCK certificate

To integrate an HSM on Demand Service with the Microsoft HCK, you must use the Luna Cryptographic Services for Windows to generate the certificate. The certificate must be signed and the signer certificate must be in the Trusted Root Certificate Authority. There are two methods to generate the file.

Method 1

1. Create an inf file with the following attributes:

```
[Version]
```

```
Signature="$Windows NT$"
[NewRequest]
Subject = "C=US,O=SafeNet,CN=HCK,OU=HCKIntegration"
KeySpec = 1
KeyLength = 2048
Exportable = FALSE
MachineKeySet = TRUE
KeyContainer = HCK
ProviderName = "Luna Cryptographic Services for Microsoft Windows"
ProviderType = 1
KeyUsage = 0x04
```

2. Generate a certificate request using the **.inf** file. Ensure you use the 32 bit certreq utility inside the **C:\Windows\SysWOW64** directory.
3. Have the certificate signed by a trusted certificate authority.
4. Import the signed certificate into the user's personal store. Ensure you select the 32 bit Microsoft Certificate Manager Console.

   C:\Windows\SysWOW64\certmgr.msc
5. Right-click **Personal**, select **All Tasks** and click **Import**. Follow the procedure to import the signed certificate.
6. Double-click the certificate and confirm that there is a private key mapped to the certificate.
7. If the certificate is not mapped correctly you can repair it using the **certutil –repairstore** command.
8. Open the certificate. Open the **Details** tab and select the **Serial Number** field. Copy the serial number.
9. Execute **certutil -repairstore -user My** <serial_number> from the **\SysWOW64** directory.

Method 2

1. Generate a certificate using Luna Cryptographic services.

   C:> makecert –sk <key_container_name> **-sp** <API_provider_name> **–r –n CN=**<certificate_name> **-ss** <name_of_certificate_store> <certificate_name>**.cer**
2. Navigate to **C:\Windows\SysWOW64** and open the **certmgr.msc**.
3. Import the certificate into the Trusted Root Certificate Authority folder. Double-click the certificate and confirm that there is a private key mapped to the certificate.
4. Open Windows Hardware Certification Kit and import the project to sign.
5. Verify the project imported correctly.
6. Go to the package tab and click on **Create package** to sign the package. When the Signing Options Window displays enable the **Use Certificate Store** radio button. Click **OK**.
7. Select the signing certificate. When the Windows Hardware Certification Package Signing Window displays select the certificate that was recently imported. Click **OK**.
8. Select a location to save the signed package. Click **Save**.
9. The Creating Package Window will display. If the certificate and the private key are correctly mapped, a success message displays and you can verify the signed package in the location you saved it.